UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/088,988	07/31/2002	Xiaoning Nie	1406/52	9022

25297 7590 05/31/2006

JENKINS, WILSON, TAYLOR & HUNT, P. A.
3100 TOWER BLVD
SUITE 1200
DURHAM, NC 27707

EXAMINER	
RIZZUTO, KEVIN P	

ART UNIT	PAPER NUMBER
2183	

DATE MAILED: 05/31/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/088,988	NIE, XIAONING	
	Examiner	Art Unit	
	Kevin P Rizzuto	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *3/21/06*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1 and 3* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1 and 3* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on ____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date 3/21/06
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1 and 3 have been examined.

2. Acknowledgement of papers filed: amendment on 3/21/06 the papers filed have been placed on record.

Claim Interpretation

3. Applicant's attention is directed towards MPEP 2114 [R-1] "Apparatus and Article Claims — Functional Language", which is copied below for Applicant's convenience. Claim 3 is an apparatus claim that contains multiple instances of functional language. However, these features do not structurally distinguish a claimed invention from prior art, and thus do not define patentable differences. For example, claim 3 states, "an instruction decoder *for decoding a processor instruction that contains an instruction opcode*..." and also, "a post condition, *"which specifies that a conditional jump is to be processed and the corresponding flag bits of an arithmetic-logic unit are to be checked*." (Step a)

MPEP 2114 [R-1] Apparatus and Article Claims — Functional Language

APPARATUS CLAIMS MUST BE STRUCTUR-ALLY DISTINGUISHABLE FROM THE PRIOR ART

>While features of an apparatus may be recited either structurally or functionally, claims< directed to >an< apparatus must be distinguished from the prior art in terms of structure rather than function. >In re Schreiber, 128 F.3d 1473, 1477-78, 44 USPQ2d 1429,1431-32 (Fed. Cir. 1997) (The absence of a disclosure in a prior art reference relating to function did not defeat the Board's finding of anticipation of claimed apparatus because the limitations at issue were found to be inherent in the prior art reference); see also In re Swinehart, 439 F.2d 210, 212-13, 169 USPQ 226, 228-29 (CCPA 1971);< In re Danly, 263 F.2d 844, 847, 120 USPQ 528, 531 (CCPA 1959). "[A]pparatus claims cover what a device is, not what a device does." Hewlett-Packard Co. v. Bausch & Lomb Inc., 909 F.2d 1464, 1469, 15 USPQ2d 1525, 1528 (Fed. Cir. 1990) (emphasis in original).

MANNER OF OPERATING THE DEVICE DOES NOT DIFFERENTIATE APPARATUS CLAIM FROM THE PRIOR ART

A claim containing a "recitation with respect to the manner in which a claimed apparatus is intended to be employed does not differentiate the claimed apparatus from a prior art apparatus" if the prior art apparatus teaches all the structural limitations of the claim. Ex parte Masham, 2 USPQ2d 1647 (Bd. Pat. App. & Inter. 1987) (The preamble of claim 1 recited that the apparatus was "for mixing flowing developer material" and the body of the claim recited "means for mixing..., said mixing means being stationary and completely submerged in the developer material". The claim was rejected over a reference which taught all the structural limitations of the claim for the intended use of mixing flowing developer. However, the mixer was only partially submerged in the developer material. The Board held that the amount of submersion is immaterial to the structure of the mixer and thus the claim was properly rejected.)

New Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. Claims 1-3 are rejected under 35 U.S.C. 103(a) as being unpatentable over Auslander, EPO 0 130 381, in view of Keneko, U.S. Patent 4,907,192, and further in view of Mahlke et al., "A Comparison of Full and Partial Predicated Execution Support for ILP Processors", herein referred to as Mahlke.

6. As per claim 1, Auslander teaches a method for processing conditional instructions in a processor with pipeline architecture, the method comprising:

a. Loading and decoding a processor the processor instruction containing an instruction opcode (bits 0-6), register addresses (bits 11-16, RA), a relative jump distance (bits 16-31, D field), and a post-condition (bits 6-11, BI), which specifies that a conditional jump is to be processed and the corresponding flag

bit of an arithmetic-logic unit is to be checked, wherein the post-condition comprises a plurality of post-condition bits that are checked in the processor: [Fig. 4 and page 33, both the Branch True and Branch False, D-form, instructions. The post-condition bits (BI) specify a corresponding flag since the BI field is sent to the Bit Select Decode unit 60, and specifies to the Bit Select Decode unit 60 to check the corresponding flag bit from the specified register (either CR or general purpose specified by RA bits) via the AND MASK 70. The BI field is made up of a plurality of bits, which are checked (by Bit Select Decode 60, fig. 4), based on the result of the checking of the corresponding flag bit is used to determine if the conditional jump is to be processed. Therefore, the post-condition bits specify whether or not a conditional jump is to be processed and specifies a corresponding flag bit to be checked.]

b. And jumping to a jump address as a function of the relative jump distance contained in the processor instruction if the post-condition is fulfilled and the checked flag bits are set: [Page 33, if the post-condition is fulfilled when the specified corresponding flag is checked and the flag is set accordingly, at which point the branch is taken, i.e., the PC is updated by adding the D field to the current PC.]

7. While Auslander teaches a plurality of post-condition bits that are checked (to determine which corresponding flag bit to check), Auslander fails to teach wherein there are *multiple* corresponding flag bits that are checked. Furthermore, Auslander fails to teach wherein the instruction contains a precondition, which specifies under which

conditions the instruction is actually to be executed, and the step of the execution of the decoded processor instruction if the precondition is fulfilled.

8. Kaneko teaches that a system, such as described by Auslander, has a degradation of performance when there is a desire to check multiple condition bits. For instance, when determining a "greater than or equal" condition. In the system of Keneko, a greater-than or equal condition requires checking the SF and OF bits. (Keneko, Col. 3, lines 18-49, col. 9, lines 6-34, including Table 1, and col. 10, lines 29-67) Likewise, in the system of Auslander, a "greater than or equal" condition is not addressed, however, there are two flags, a "greater than" flag and an "equal flag" (see table 1(a), page 20), therefore, one of ordinary skill in the art would recognize that if either one was set, a "greater than or equal to" condition would be satisfied. Keneko teaches adding a condition selector to receive all flag bits and based on the instruction specified condition bits, to perform the appropriate checks of the corresponding *plurality of flags* (such as SF and OF for the "greater-than or equal to" condition), see fig. 5, col. 9, lines 6-34, including Table 1, and col. 10, lines 29-67. Keneko teaches that this reduces the amount of instructions to be processed and improves processor performance. (Col. 3, line 50 to col. 4, line 11.)

9. It would have been obvious to add the "condition selector" hardware to enable the processor to efficiently execute multi-condition branch instructions (branch instructions with more than one corresponding flag), such as a branch instruction with a "greater than or equal to" condition. The improved processor performance and

reduction in the number of instructions to execute would have provided motivation to one of ordinary skill in the art to combine the inventions as described above.

10. However, Auslander, in view of Keneko, fails to teach wherein the instruction contains a precondition, which specifies under which conditions the instruction is actually to be executed, and the step of the execution of the decoded processor instruction if the precondition is fulfilled.

11. Mahlke teaches wherein every instruction contains an additional source operand to hold a predicate specifier (precondition) (Page 139, left column, lines 1-3). If the precondition is true, the instruction is executed, however if it is false, the instruction is not executed. (Page 139, right column, 3rd full paragraph). Mahlke also teaches that using the predicated method in place of a large portion of branch instructions (i.e., not all) improves processor performance significantly (Page 139, left column, first full paragraph). It is therefore inherent that there are some unconverted branch instructions that contain a predicate specifier (precondition). Also, a predicated branch instruction is shown in figure 3, under heading 'fully predicated code' and sub-heading 'branch instructions'. Predicating instructions is a well-known method in the art that allows improved processing by not stalling the pipeline while a branch instruction is evaluated. Instead, the instructions that are dependent on the branch instruction contain predicates, and they are executed as normal, except their results are not committed. When the predicate contains a valid value, the instructions from the correct execution path are committed while the other instructions and results are simply ignored. (See 1 Introduction, Mahlke, page 138).

12. The combination of the full predication of Mahlke and Auslander would result in a branch instruction (conditional jump instruction) with a precondition (predicate specifier) and post-condition (c). It would have been obvious to one of ordinary skill in the art to add the full predication of Mahlke to the instruction processing of Auslander because of the improved processing performance it offers. (Mahlke, page 138, Abstract, final sentence).

13. As per claim 3, Auslander teaches an apparatus for processing conditional jump instructions in a processor with pipeline computer architecture, the apparatus comprising:

-An instruction decoder (paragraph 3, left column, page 5) for decoding a processor instruction (Branch True or Branch False instruction, page 33) that contains an instruction opcode (bits 0-6), register addresses (bits 11-16, 'RA' field), relative jump distance (bits 16-31, "D" field), and a post-condition, which specifies that a conditional jump is to be processed and the corresponding flag bit of an arithmetic-logic unit is to be checked:('bits 6-11, "BI" field): [Page 33]

-Wherein the instruction decoder (BR/TRAP Testing unit 52, page 40, line 33 to page 41, line 3 and fig. 2B) is operable to check, whether the post-condition is fulfilled and the flag bit is set, if positive driving a program counter for forming a jump address as a function of the relative jump distance contained in the processor instruction: [Page 33, if the post-condition is fulfilled when the specified bit is checked, the branch is taken, i.e., the PC is updated by adding the D field to the current PC.]

14. While Auslander teaches a plurality of post-condition bits that are checked (to determine which corresponding flag bit to check), Auslander fails to teach wherein there are *multiple* corresponding flag bits that are checked. Furthermore, Auslander fails to teach the processor instruction containing a precondition, which specifies under which conditions the instruction is actually to be executed, and the instruction decoder is operable to check, in the case of a fulfilled precondition, whether the post-condition is fulfilled and, in the case of a fulfilled post-condition, driving a program counter for forming jump address as a function of the relative jump distance contained in the processor instruction.

15. Kaneko teaches that a system, such as described by Auslander, has a degradation of performance when there is a desire to check multiple condition bits. For instance, when determining a "greater than or equal" condition. In the system of Keneko, a greater-than or equal condition requires checking the SF and OF bits. (Keneko, Col. 3, lines 18-49, col. 9, lines 6-34, including Table 1, and col. 10, lines 29-67) Likewise, in the system of Auslander, a "greater than or equal" condition is not addressed, however, there are two flags, a "greater than" flag and an "equal flag" (see table 1(a), page 20), therefore, one of ordinary skill in the art would recognize that if either one was set, a "greater than or equal to" condition would be satisfied. Keneko teaches adding a condition selector to receive all flag bits and based on the instruction specified condition bits, to perform the appropriate checks of the corresponding *plurality of flags* (such as SF and OF for the "greater-than or equal to" condition), see fig. 5, col. 9, lines 6-34, including Table 1, and col. 10, lines 29-67. Keneko teaches that this

reduces the amount of instructions to be processed and improves processor performance. (Col. 3, line 50 to col. 4, line 11.)

16. It would have been obvious to add the "condition selector" hardware to enable the processor to efficiently execute multi-condition branch instructions (branch instructions with more than one corresponding flag), such as a branch instruction with a "greater than or equal to" condition. The improved processor performance and reduction in the number of instructions to execute would have provided motivation to one of ordinary skill in the art to combine the inventions as described above.

17. However, Auslander, in view of Keneko, fails to teach wherein the instruction contains a precondition, which specifies under which conditions the instruction is actually to be executed, and the step of the execution of the decoded processor instruction if the precondition is fulfilled.

18. Mahlke teaches wherein every instruction contains an additional source operand to hold a predicate specifier (precondition) (Page 139, left column, lines 1-3). If the precondition is true, the instruction is executed, however if it is false, the instruction is not executed. (Page 139, right column, 3rd full paragraph). Mahlke also teaches that using the predicated method in place of a large portion of branch instructions (i.e., not all) can improve processor performance significantly (Page 139, left column, first full paragraph). It is therefore inherent that there are some unconverted branch instructions that contain a predicate specifier (precondition). Also, a predicated branch instruction is shown in figure 3, under heading 'fully predicated code' and sub-heading 'branch instructions'. Predicating instructions is a well-known method in the art that allows

improved processing by not stalling the pipeline while a branch instruction is evaluated. Instead, the instructions that are dependent on the branch instruction use predicates, and the instructions are executed as normal, except their results are not committed. When the predicate contains a valid value, the instructions from the correct execution path are committed while the other instructions and results are ignored or discarded. (See 1 Introduction, Mahlke, page 138).

19. The combination of the full predication of Mahlke and Auslander would result in a branch instruction (conditional jump instruction) with a precondition (predicate specifier) and post-condition (BI), wherein the precondition would prevent the conditional jump instruction from being executed if false. It would have been obvious to one of ordinary skill in the art to add the full predication of Mahlke to the instruction processing of Auslander because of the improved processing performance it offers. (Mahlke, page 138, Abstract, final sentence).

Response to Arguments

1. Applicant's arguments regarding the amendment portion filed on 3/21/06 have been fully considered and are found persuasive. Specifically, Auslander fails to teach that the post-condition bits specify that multiple corresponding flag bits are to be checked. Thus, a new 35 U.S.C. 103 Rejection is provided above. However, certain arguments are still considered pertinent and are addressed below.

2. Applicant argues the novelty/rejection of claims 1 and 3.

"Auslander neither teaches nor suggests a post-condition which specifies that a conditional jump is to be processed and the corresponding flag bits of an arithmetic-logic unit are to be checked."

"Furthermore, the present invention is distinguished from Auslander by the present amendments reciting that the post-condition comprises a plurality of post-condition bits that are checked in the processor."

3. These arguments are not found persuasive for the following reasons:

a. To clarify, applicant's attention is directed towards Auslander, fig. 4 and page 33. The BI field is equated to the claimed "post-condition bits". The BI field is made up of a plurality of bits, which are sent to the Bit Select Decode 60. The Bit Select Decode 60 "checks" the bits, since it must appropriately output a value along bus 72 to the AND MASK 70 based on the BI field's value, i.e., the Bit Select Decode 60 checks the bits, and dependent on their value, outputs an appropriate value on bus 72. Furthermore, the post-condition bits specify that a conditional jump is to be processed by specifying a condition to check (corresponding flag), which dependent on the instruction and it's setting, will cause a jump to be processed. Thus, each limitation regarding the post-condition bits is taught by Auslander. While there may be differences between the teachings of Auslander and the invention as described in Applicant's specification, the claimed invention has been taught by Auslander, in view of Kaneko and Mahlke as set forth above.

Conclusion

The following is text cited from 37 CFR 1.111(c): In amending in reply to a rejection of claims in an application or patent under reexamination, the applicant or patent owner must clearly point out the patentable novelty which he or she thinks the

claims present in view of the state of the art disclosed by the references cited or the objections made. The applicant or patent owner must also show how the amendments avoid such references or objections.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Kevin P Rizzuto whose telephone number is (571) 272-4174. The examiner can normally be reached on M-F, 8-4:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

KPR

EDDIE CHAN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100